Project Report

On

Image Cryptography with RSA

Submitted In The Partial Fulfillment Of The Requirements For The Award Of The Degree OF

B.Tech

IN

COMPUTER SCIENCE & ENGINEERING

Under Supervision of:
Er. Manpreet Kaur

Submitted by:
Nishant Bharti (15110185)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

GIANI ZAIL SINGH COLLEGE CAMPUS OF ENGINEERING &TECHNOLOGY, BATHINDA-151001

ACKNOWLEDGMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project. Whatever we have done is only due to such guidance and assistance and we would not forget to thank them.

We express our deepest gratitude to ***Dr. Naresh Kumar Garg*** worthy HOD for extending his inevitable and valuable support to us.

We owe our profound gratitude to our **guide** for the project **Ms. Manpreet Kaur** who took keen interest in our project work and guided us all along till the completion of our project work by providing all the necessary information for developing a good system. She has taken immense effort and pain to go through the project and make necessary corrections as and when needed.

We are thankful to and fortunate enough to get constant encouragement, support and guidance from our faculty members without whom this project would have been a distant reality. We would also like to extend our sincere regards to our families, and well wishers for their timely support.

NISHANT BHARTI

DECLARATION

We hereby declared that work presented in work entitled **Image Cryptography with RSA** in partial fulfillment of the requirements for the award of degree of bachelor of computer application submitted in **"***Giani Zail Singh Campus College of Engg. &Tech.***"** affiliated to **Maharaja Ranjit Singh Punjab Technical University** is an authentic record of my own work carried out under the supervision of Ms**. Manpreet Kaur.**

Nishant Bharti(15110185)

Place: BATHINDA

Date: 15-05-2019

CONTENT PAGE

LIST OF FIGURES :

CHAPTER-1

INTRODUCTION

INTRODUCTION

1.1 Company Profile :

Sunsoft is an emerging software development company. With over 5 years of experience in the industry, we understand what it takes to produce successful corporate image. The right image embodies the company's fundamental essence and at the same time increases the customer's identification with the brand.

The Sunsoft is an image development company that comprises of business, marketing and graphics experts who deliver innovative and eye-catching designs to globally positioned firms.

We use the most up to date visual communication and technology in order to combine the company culture, values and vision in an innovative and outstanding image.

TEAM: We have a committed team of professionals with experience in variety of tools and platforms. Our assistants keep themselves updated with the latest technology trends. Our skilled people share a common vision of growth. We do what we believe in, and we believe in what we do.

BELIEF: We believe in utilizing technology to make things simple and easy to use. We also believe in thinking ahead and delivering training about technology that can fit well with upcoming future technologies.

VALUES: We believe that the rich learning experiences our programs offer can effect powerful and positive transformation in individuals, in their communities, and in their institutions. We work with our partners and sponsors worldwide to design and implement programs in which participants reach across – and live across – boundaries and borders. And we do this in an atmosphere of mutual respect, organized collaboration, and fun.

FACILITIES:Some of the facilities offered in SUNSOFT are as follows:

- Trainers at the centre have 5 years of experience in all languages.
- To ensure the quality, all the trainers are certified in their own domain .
- For learning to be comprehensive, the requirement is to automate complex scenarios. Project assignments are given to students to help them realize real time experience.

1.2 JAVA:

FIG 1.1 LOGO OF JAVA

Java is an object-oriented programming language with a built-in application programming interface (API) that can handle graphics and user interfaces and that can be used to create applications or applets. Because of its rich set of API's, similar to Windows, and its platform independence, Java can also be thought of as a platform in itself. Much of the syntax of Java is the same as C and C++. One major difference is that Java does not have pointers.

Java applications are typically compiled to byte code that can run only on Java virtual machine (JVM) regardless of computer architecture.

In Java we distinguish between applications, which are programs that perform the same functions as those written in other programming languages, and applets, which are programs that can be embedded in a Web page and accessed over the Internet. Our initial focus will be on writing applications. When a program is compiled, a byte code is produced that can be read and executed by any platform that can run Java.

There were five primary goals in the creation of the java language:

- It should be "simple, object-oriented and familiar"
- It should be "robust and secure"
- It should be "architecture-neutral and portable"
- It should execute with "high performance"
- It should be "interpreted, threaded, and dynamic"

1.2.1. LIBRARIES:

- java.lang: Provides classes that are fundamental to the design of the Java programming language. The most important classes are Object, which is the root of the class hierarchy, and Class, instances of which represent classes at run time.
- java.io**:** Provides for system input and output through data streams, serialization and the file system. Unless otherwise noted, passing a null argument to a constructor or method in any class or interface in this package will cause a NullPointerException to be thrown.
- java.awt: Contains all of the classes for creating user interfaces and for painting graphics and images. A user interface object such as a button or a scrollbar is called, in AWT terminology, a component. The Component class is the root of all AWT components. See Component for a detailed description of properties that all AWT components share.
- javax.swing: Provides a set of "lightweight" (all-Java language) components that, to the maximum degree possible, work the same on all platforms..
- Java.imageio: A class containing static convenience methods for locating Image Readers and Image Writers, and performing simple encoding and decoding.
- Java.util: Contains the collections framework, legacy collection classes, event model, date and time facilities, internationalization, and miscellaneous utility classes (a string tokenizer, a random-number generator, and a bit array)
- Javax.swing.JFileChooser: The class **JFileChooser** is a component which provides a simple mechanism for the user to choose a file.
- Javax.swing.JOptionPane: The JOptionPane class is used to provide standard dialog boxes such as message dialog box, confirm dialog box and input dialog box. These dialog boxes are used to display information or get input from the user.
- Javax.swing.filechooser.FileNameExtensionFilter: An implementation of FileFilter that filters using a specified set of extensions. The extension for a file is the portion of the file name after the last ".". Files whose name does not contain a "." have no file name extension. File name extension comparisons are case insensitive.
- Java.awt.image.BufferedImage: The BufferedImage subclass describes an image with an accessible buffer of image data. A BufferedImage is comprised of a ColorModel and a Raster of image data. The number and types of bands in the SampleModel of the Raster must match the number and types required by the ColorModel to represent its color and alpha components. All BufferedImage objects have an upper left corner coordinate of (0, 0).

- Java.awt.Color: The Color class is used to encapsulate colors in the default s RGB color space or colors in arbitrary color spaces identified by a ColorSpace. Every color has an implicit alpha value of 1.0 or an explicit one provided in the constructor. The alpha value defines the transparency of a color and can be represented by a float value in the range 0.0 - 1.0 or 0 - 255. An alpha value of 1.0 or 255 means that the color is completely opaque and an alpha value of 0 or 0.0 means that the color is completely transparent.
- Java.awt.Graphics2D: This Graphics2D class extends the Graphics class to provide more sophisticated control over geometry, coordinate transformations, color management, and text layout. This is the fundamental class for rendering 2-dimensional shapes, text and images on the Java(tm) platform.
- Java.io.IOException: Signals that an I/O exception of some sort has occurred. This class is the general class of exceptions produced by failed or interrupted I/O operations.

1.2.2. JAVA RUNTIME ENVIRONMENT (JRE) :

The Java Runtime Environment (JRE) is a set of software tools for development of Java applications. It combines the Java Virtual Machine (JVM), platform core classes and libraries.

JRE is part of the Java Development Kit (JDK), but can be downloaded separately. JRE was originally developed by Sun Microsystems Inc., a wholly owned subsidiary of Oracle Corporation. Also known as Java runtime.

JRE consists of the following components:

1. Deployment technologies, including deployment, Java Web Start and Java Plug-in.
2. User interface toolkits, including Abstract Window Toolkit (AWT), Swing, Java 2D, Accessibility, Image I/O, Print Service, Sound, drag and drop (DnD) and input methods.
3. Integration libraries, including Interface Definition Language (IDL), Java Database Connectivity (JDBC), Java Naming and Directory Interface (JNDI), Remote Method Invocation (RMI), Remote Method Invocation Over Internet Inter-Orb Protocol (RMI-IIOP) and scripting.
4. Other base libraries, including international support, input/output (I/O), extension mechanism, Beans, Java Automation Extensions (JMX), Java Native Interface (JNI), Math, Networking, Override Mechanism, Security, Serialization and Java for XML Processing (XML JAXP).

1.2.3. JAVA VIRTUAL MACHINE (JVM) :

A Java virtual machine (JVM) is an abstract computing machine. There are three notions of the JVM: specification, implementation, and instance. An instance of the JVM can execute any executable computer program compiled into Java bytecode. It is the code execution component of the Java platform.

In the Java programming language, all source code is first written in plain text files ending with the .java extension. Those source files are then compiled into .class files by the javaccompiler. A .class file does not contain code that is native to your processor; it instead contains bytecodes — the machine language of the Java Virtual Machine (Java VM). The java launcher tool then runs your application with an instance of the Java Virtual Machine.

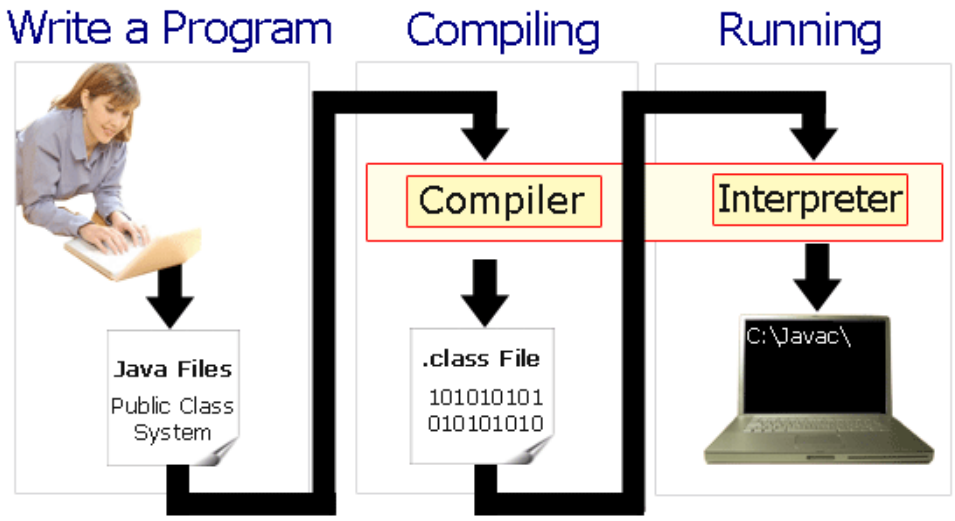


FIG.1.2: Compilation and Interpretation of Java

Because the Java VM is available on many different operating systems, the same .class files are capable of running on Microsoft Windows, the Solaris™ Operating System (Solaris OS), Linux, or Mac OS. Some virtual machines, such as the Java SE HotSpot at a Glance, perform additional steps at runtime to give your application a performance boost. This include various tasks such as finding performance bottlenecks and recompiling (to native code) frequently used sections of cod

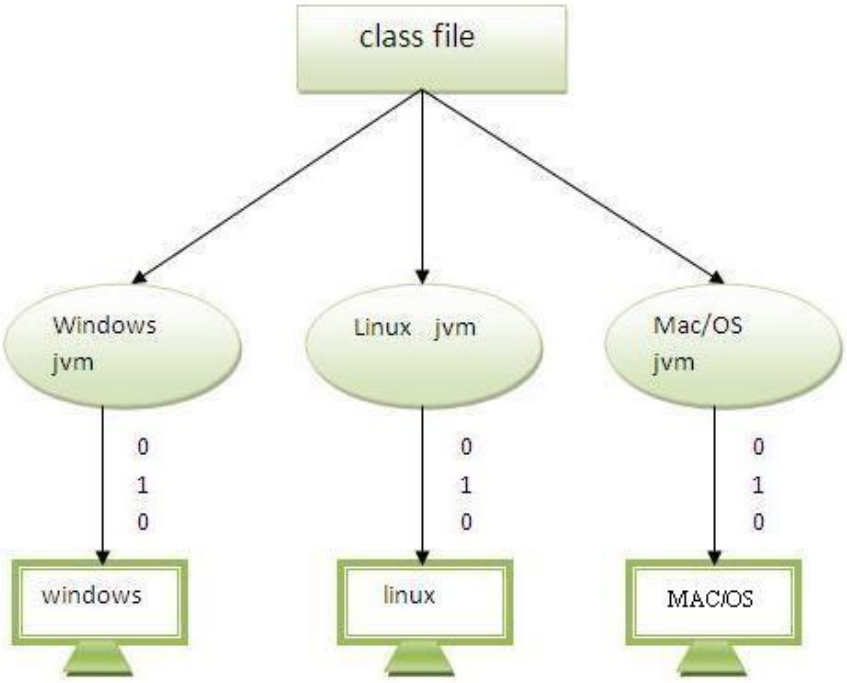


Fig 1.3: Platform Independence of java

1.2.4. FEATURES OF JAVA :

- Platform Independent.:

  A platform is the hardware or software environment in which a program runs. There are two types of platforms software-based and hardware-based. Java provides software-based platform. The Java platform differs from most other platforms in the sense that it's a software-based platform that runs on top of other hardware-based platforms. It has two components:

  1. Runtime Environment
  2. API(Application Programming Interface)

  Java code can be run on multiple platforms e.g. Windows, Linux, Sun Solaris, Mac/OS etc. Java code is compiled by the compiler and converted into bytecode . This bytecode is a platform independent code because it can be run on multiple platforms i.e. Write Once and Run Anywhere (WORA).

- Robust and Secure :

  Robust simply means strong. Java uses strong memory management. There are lack of pointers that avoids security problem. There is automatic garbage collection in java. There is exception handling and type checking mechanism in java. All these points makes java robust.

- Distributed :

We can create distributed applications in java. RMI and EJB are used for creating distributed applications. We may access files by calling the methods from any machine on the internet.

- Multithreaded and Interactive :
  A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it shares the same memory. Threads are important for multi-media, Web applications etc.

- Dynamic and Extensible Code :
  Java is faster than traditional interpretation since byte code is "close" to native code still somewhat slower than a compiled language (e.g., C++).
- Architectural Neutral :
  There is no implementation dependent features e.g. size of primitive types is set.

1.3 FRAMEWORK AND SOFTWARE USED:

1.3.1 SWINGS :

Swing implements a set of GUI components that build on AWT technology and provide a pluggable look and feel. Swing is implemented entirely in the Java programming language, and is based on the JDK 1.1 Lightweight UI Framework.

Swing features include:

- All the features of AWT.
- 100% Pure Java certified versions of the existing AWT component set (Button, Scrollbar, Label, etc.).
- A rich set of higher-level components (such as tree view, list box, and tabbed panes).
- Pure Java design, no reliance on peers.
- Pluggable Look and Feel.

Swing components do not depend on peers to handle their functionality. Thus, these components are often called "lightweight" components.

Pros

- Portability: Pure Java design provides for fewer platform specific limitations.
- Features: Swing supports a wider range of features like icons and pop-up tool-tips for components.
- Vendor Support: Swing development is more active. Sun puts much more energy into making Swing robust.
- Look and Feel: The pluggable look and feel lets you design a single set of GUI components that can automatically have the look and feel of any OS platform (Microsoft Windows, Solaris, Macintosh, etc.).

Cons

1. Applet Portability: Most Web browsers do not include the Swing classes, so the Java plugin must be used.
2. Performance: Swing components are generally slower and buggier than AWT, due to both the fact that they are pure Java and to video issues on various platforms. Since Swing components handle their own painting (rather than using native API's like DirectX on Windows) you may run into graphical glitches.

1.3.2 NetBeans IDE 8.2 :

NetBeans IDE is a free, open source, integrated development environment (IDE) that enables you to develop desktop, mobile and web applications. The IDE supports application development in various languages, including Java, HTML5, PHP and C++. The IDE provides integrated support for the complete development cycle, from project creation through debugging, profiling and deployment. The IDE runs on Windows, Linux, Mac OS X, and other UNIX-based systems.

The IDE provides comprehensive support for JDK 8 technologies and the most recent Java enhancements. It is the first IDE that provides support for JDK 8, Java EE 7, and JavaFX 2. The IDE fully supports Java EE using the latest standards for Java, XML, Web services, and SQL and fully supports the GlassFish Server, the reference implementation of Java EE.

System Requirements

NetBeans IDE runs on operating systems that support the Java VM (Virtual Machine) and has been tested on the platforms listed below.

Note: The IDE's minimum screen resolution is 1024x768 pixels.

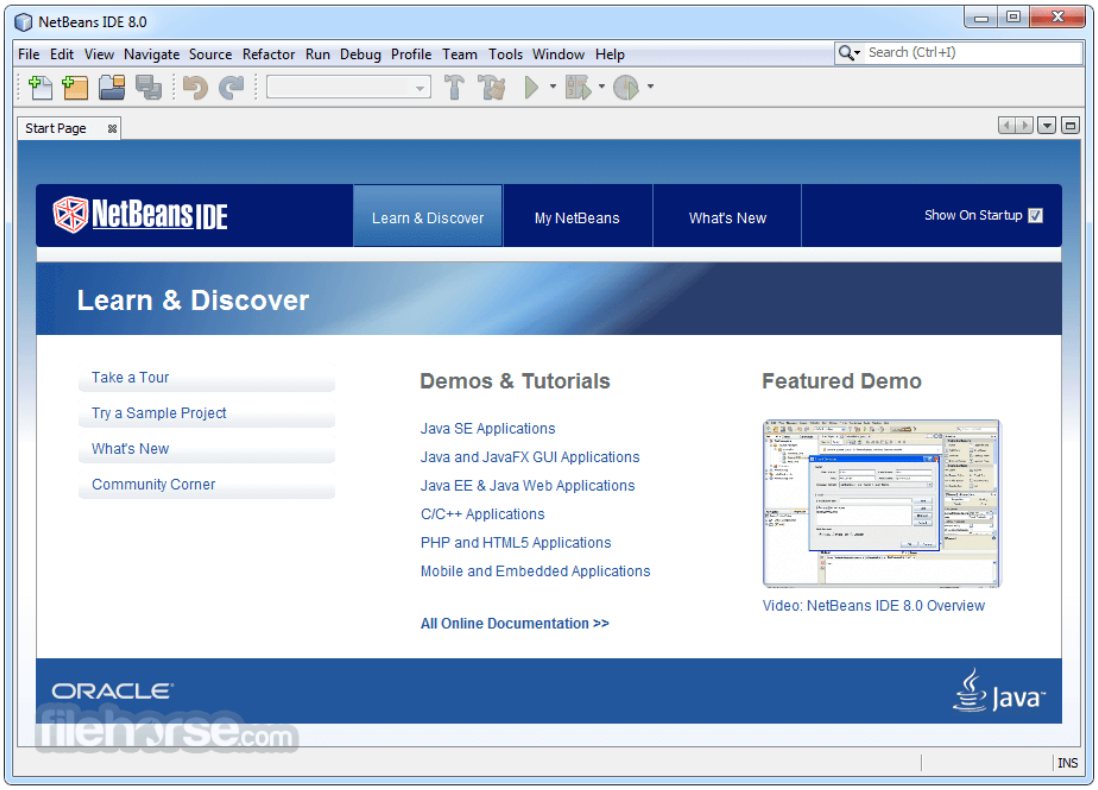
Supported Operating Systems

Minimum Hardware Configurations

- **Microsoft Windows Vista SP1/Windows 7 Professional:**
  - **Processor:** 800MHz Intel Pentium III or equivalent
  - **Memory:** 512 MB
  - **Disk space:** 750 MB of free disk space

- **Ubuntu 9.10:**
  - **Processor:** 800MHz Intel Pentium III or equivalent
  - **Memory:** 512 MB
  - **Disk space:** 650 MB of free disk space

- **Macintosh OS X 10.7 Intel:**
  - **Processor:** Dual-Core Intel
  - **Memory:** 2 GB
  - **Disk space:** 650 MB of free disk space


NetBeans IDE 8.0
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
Start Page
NetBeans IDE
Learn & Discover
My NetBeans
What's New
Show On Startup
Learn & Discover
Take a Tour
Try a Sample Project
What's New
Community Corner
Demos & Tutorials
Java SE Applications
Java and JavaFX GUI Applications
Java EE & Java Web Applications
C/C++ Applications
PHP and HTML5 Applications
Mobile and Embedded Applications
All Online Documentation >>
Featured Demo
Video: NetBeans IDE 8.0 Overview
ORACLE
Java
INS

CHAPTER-2

INTRODUCTION TO PROJECT

2.1 PROJECT DEFINITION:

The project entitled “Image Cryptography’ is a complex project involving the conversion of image into matrix form, using a mathematical concept to encrypt and decrypt it, was instrumental in giving us a thorough understanding of how the concepts of Linear Algebra together can actually be implemented in the real world. This project plays an important role in the field of information hiding. Image encryption method prepared information unreadable. Therefore, no hacker or eavesdropper, including server administrators and others, have access to original message or any other type of transmitted information through public networks such as internet.

2.2 DESCRIPTION OF PROJECT:

Nowadays, information security is becoming more important in data storage and transmission. Images are widely used form of data transmission. Therefore, the security of image data from unauthorized user is more important. Text prefers more data to be sent relatively than the simple character text.

2.2.1 Existing System:

The existing system inputs the image to be encrypted from the user and then perform the encryption algorithm over it. The resultant key is than displayed over the screen to the user for further decryption reference. On the other hand, the decryption is done by choosing the encrypted image and then applying the decryption algorithm. This algorithm only works when the needed cryptography-key is provided to it. The key then initiates the decryption algorithm and the digital plain image in the .png format is displayed on to the screen.

2.2.2 Limitations in Existing System:

Two main problems that arise in the image encryption process are with respect to:

- Computational time
- Security level

Real time image encryption prefer cipher that take less amount of **computational time** with compromising **security**.

2.2.3. Proposed System:

We are very excited by the vast future possibilities that our project has to offer. In the proposed system, we would like to add the following implementations.

The cryptography-key generated by this algorithm is long enough to be remembered by a human being unless he/she writes up that key somewhere for further reference. The key is the basic thing which keeps up this whole setup. If the key is lost or stolen, the data is lost. The new key generating algorithm should be used which actually takes up the “cryptography-key” from the user in the form of password and encrypts that image in such a way that satisfies that given cryptography-key given. The decryption should be done the same instead of providing the cryptography-key, user provides the password to decrypt that image. But for this implementation the algorithm should be strong

enough to take up any entry as its cryptography-key. Further to encrypt the image according to that password is a tough job. We are also looking forward to encrypt videos by extracting each frame and encrypting the images simultaneously. We know that all the videos have sound. So we are planning to encrypt frames and sound simultaneously. Finally after achieving all of the above, we are planning to create an app which will do all of the above. With two people having the app, one will become the sender and other the receiver at a time, based on the requirements of either of the two. This is future of our project we are looking at and looking forward to implementing all of the above successfully.

2.2.4. Advantages over Existing System:

The advantages of the Image Cryptography system are:

1. Easy to implement anywhere at any system. Whether we are working on Windows or on Linux, we can use this desktop application at any platform.
2. User Interface is not at all complex to use. Any user with some basic computer knowledge understanding can perform the encryption and decryption methods presented in the application.
3. This images produced after the encryption results in the high quality .png encrypted files and the decrypted image also results in the .png file which results in the better quality than the .jpg or .jpeg image files.

CHAPTER-3

SCREENSHOTS

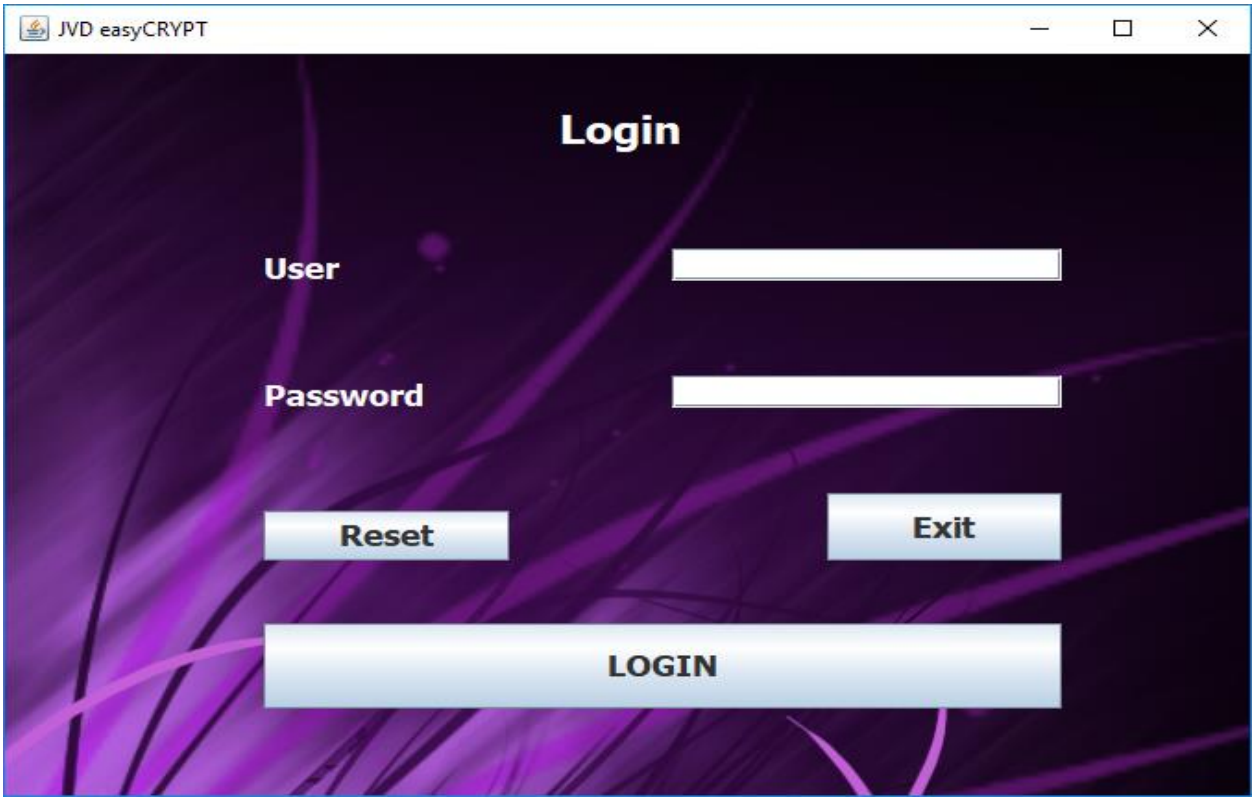


Fig. 3.1 Image Cryptography

Fig. 3.2 Portal

Fig. 3.3 Encrption Portal

Fig. 3.4 File Selection

Fig. 3.5 Successfully Encrypted

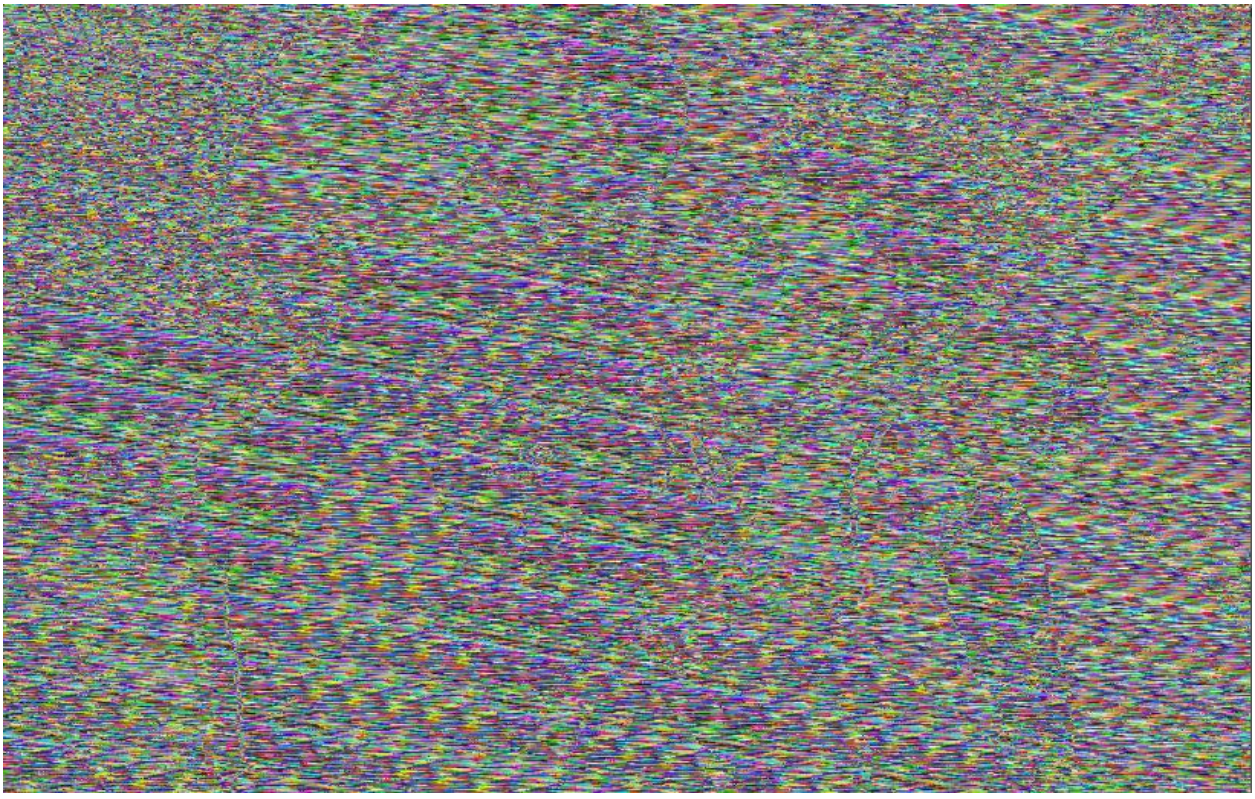


Fig. 3.6 Encrypted File

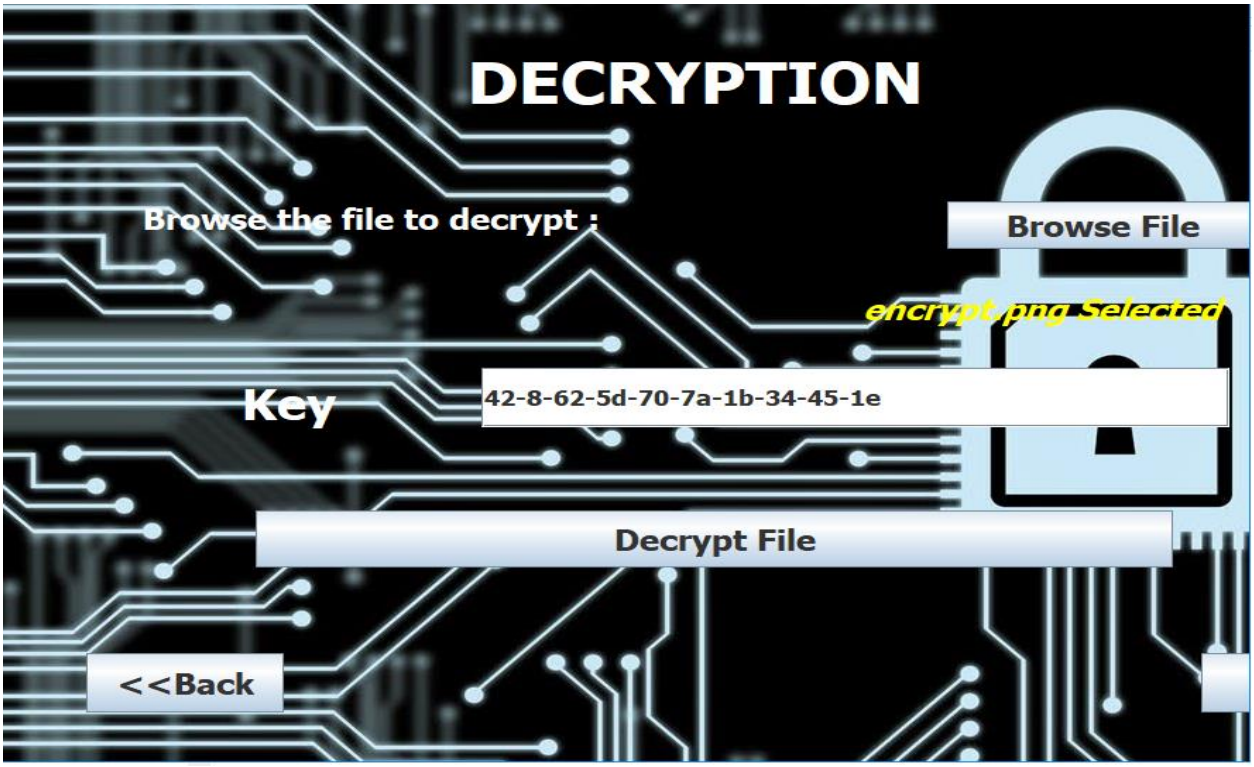


Fig. 3.7 Decryption Portal

Fig. 3.8 Successfully Decrypted

CHAPTER-4

CONCLUSION AND FUTURE SCOPE

CONCLUSION AND FUTURE SCOPE

Image Cryptography with RSA software where Admin has the rights to add/view/delete accountant and Accountant has the rights to add/view /edit/delete student ,check due fees and logout. So basically there are two users we have used. This is a simple software with no use of special effects using css , jsp , servlet and etc . So, it is a simple software that can be used in coaching institutions and private institutions where the accountant needs to know about the student related information and about the fees of the individual student. There are more than one accountants. There is also an option of updating the records of the student. It is having no special effects but completing all necessary requirements. In which the accountant knows about how all the information about the student, the moment the user applies there either by reaching the website or going to the venue. The Front end has been developed in Java Swing that develops windows based application and the Back End has been developed in MySQL to import sql files to create tables in MySQL. It can be used in various in private institutions, training institutes, and coaching centers. It increases the employment of the accountants. Hundreds and hundreds of records of the students can be stored . But the software is not so capable to challenge other software because the software industry is using more accomplished software having more security and software could also be attractive .This can be done by using jsp , css and servlet. So, certain changes are expected to be done. Then the software will be used by most of the good training institutes.

CHAPTER-5

BIBLIOGRAPHY & REFERENCES

BIBLIOGRAPHY AND REFERENCES

The following books are used to fulfill the requirements of project and are helpful in understanding, development and the maintenance of the project.

- http://www.w3schools.com
- http://docs.oracle.com
- http://www.programmerinterview.com
- http://javatpoint.com
- www.google.co.in
- www.wikipedia.com
- www.eclipse.org
- www.studytonight.com
- http://www.jdbc-tutorial.com/
- Stackoverflow.com
- Mit opencourseware.com